AXP-033-050322

PLUG MAGIC シリーズ A/D コンバータ

AXP-AD02

取扱説明書

株式会社 アドテック システム サイエンス

ご 注 意

1. 本製品の外観や仕様および取扱説明書に記載されている事項は、将来予告なしに変更することがあります。

2. 本ソフトウェアの内容および取扱説明書に記載のすべての事項について、株式会社アドテック システム サイエンスから文書による許諾を得ずにおこなう、あらゆる複製も転載も禁じます。

3. この取扱説明書に記載されている会社名や商品名は、各社の商標および登録商標です。

4. 取扱説明書の内容を十分に理解しないまま本製品を扱うことは、絶対におやめください。本製品の取り扱いについては安全上細心の注意が必要です。取り扱い説明を十分に理解してから本製品をご使用ください。

5. 本ソフトウェアを使用した結果生じた他への影響については、いっさい責任を負いかねますので、あらかじめご了承ください。

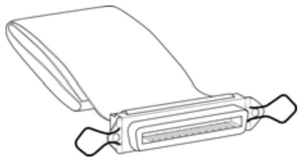
すべて揃っていますか

本体と次の付属品がすべて揃っているか確認してください。
万一、不足の品がありましたらお手数でもお買い上げの販売店もしくは当社までご連絡ください。

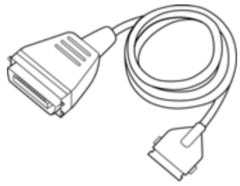
同梱品



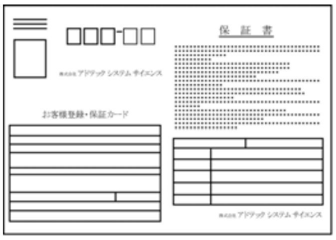
□AXP-AD02 カード本体



□付属ケーブル



□専用接続ケーブル



□お客様登録カード・保証書



□サポートディスク

《おことわり》

(1) 本書の内容の一部または全部を無断で記載することは、禁止されております。
(2) 本製品の仕様および本書の内容は、将来予告なく変更することがあります。
(3) 本書の内容につきましては、万全を期して作成いたしましたが、万一ご不審な点やお気付きの点がございましたら、当社までご連絡ください。
(4) 本製品は、出荷の際十分な検査を行い万全を期しておりますが、万一ご使用中にご不審な点がございましたら、当社までご連絡ください。
(5) 本製品につきましては、保証書に明記された条件における保証期間中の修理をもって、当社の唯一の責任とさせていただきます。本製品を運用した結果の影響につきましては、(3)(4)項にかかわらず責任を負いかねます。
(6) 本文中にある会社名、商品名は各社の商標または登録商標です。

目次

はじめに 1
取り扱い上の注意 2
1. カードの取り付け 3
1-1. ケーブルをカードに取り付ける 3
1-2. パソコンへカードを取り付ける 3
2. ソフトウェアの組み込み 4
2-1. MS-DOS で使用する場合 5
2-2. Windows95/98/Me で使用する場合 6
2-2-1. カードの登録（インストール） 6
2-2-2. 登録（インストール）の確認 9
2-3. Windows2000/XP で使用する場合 13
2-3-1. カードの登録（インストール） 13
2-3-2. 登録（インストール）の確認 17
3. アプリケーションの作成 22
3-1. サポートソフトの内容 22
3-2. I/O マップ 24
3-2-1. I/O マップ 24
3-2-2. レジスタの説明 24
3-3. MS-DOS アプリケーションの作成 31
3-3-1. サンプルソース及び関数ライブラリ概要 31
3-3-2. 関数ライブラリ解説補足 31
3-4. Windows アプリケーションの作成 33
3-4-1. サンプリング動作モード設定について 33
3-4-2. サンプリング開始トリガについて 36
3-4-3. 入力電圧レンジについて 38
3-4-4. チャンネルインクリメントサンプリングについて 40
3-4-5. サンプリングステータス監視について 41
4. 外部信号との接続 42
4-1. コネクタ信号表 42
4-2. 入出力信号等価回路 43
4-3. 信号接続例 43
5. 製品仕様 44
製品のメンテナンスについて 45
製品のお問い合わせについて 46
改訂履歴 47

はじめに

この度は、PLUG MAGIC シリーズ A/D コンバータ AXP-AD02 をお買い求めいただき、誠にありがとうございます。
本製品の性能を十分ご活用いただくため、本書を熟読され、正しい使用法で末永くご愛用いただきますようお願い申し上げます。

ー動作環境ー

■本製品は以下の動作環境でお使いください。

対応パソコン　PC Card Standard TYPEⅡ規格の PC カードスロットを持ったパソコン
DOS/V パソコン、NEC PC98 シリーズのいずれの機種も動作可能です。

! ご使用前にそのパソコンにPCカードスロットがあるかをお確かめください。

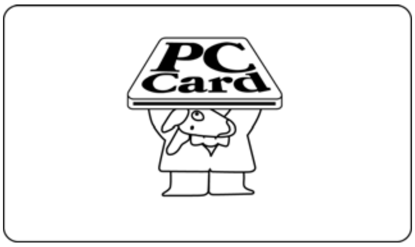
ー特　長ー

- 本製品AXP-AD02は、PC Card Standard TYPEⅡ規格に準拠したA/Dコンバータです。PCカードスロット装備の各種パソコンへ装着することにより、簡単にアナログ入力の拡張ができます。
- 逐次比較変換方式A/Dコンバータに512ワードFIFOメモリの採用と4ビット汎用入出力兼外部トリガー機能により多種多様なアプリケーションの生成が容易です。
- アナログ入力チャンネルは8CH。
- サンプリング機能に10μs～2.55msのタイマーサンプル、チャンネルオートインクリメント機能。
- 付属のコントロールプログラムにより環境に合わせたキャリブレーション及び入力レンジコントロールが可能です。また、コントロールプログラムはC言語により開発されており、個々の機能はCの関数ライブラリになっていますのでハードウェアを意識することなくアプリケーションの開発が可能です。

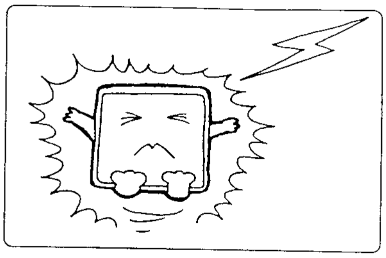
取り扱い上の注意

本製品は非常に精密な電子機器です。お取り扱いに際しては、次の事項を守ってご使用ください。

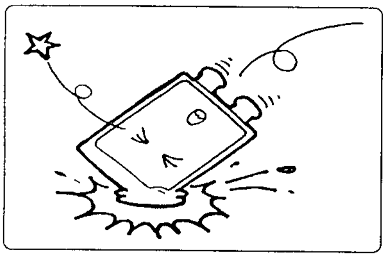
■このカードはPC Card Standard対応カードスロット以外では使用できません。



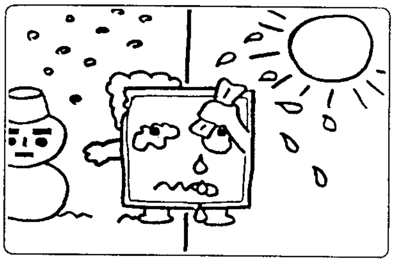
■静電気に弱いので、静電気の起きやすい場所等に放置しないでください。



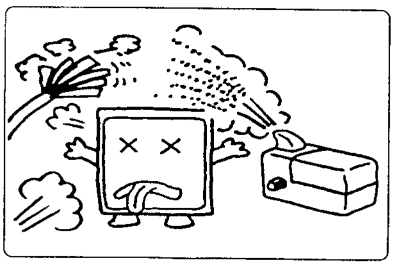
■本体に衝撃をあたえたり、落としたりしないでください。



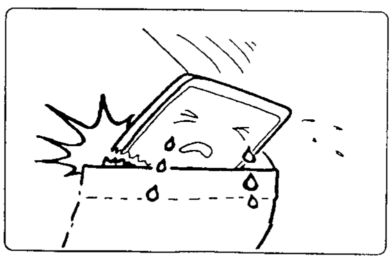
■直射日光の当たる場所や低温な場所での使用や保管は避けてください。



■ほこりや湿気の多いところでの使用や管理はさけてください。



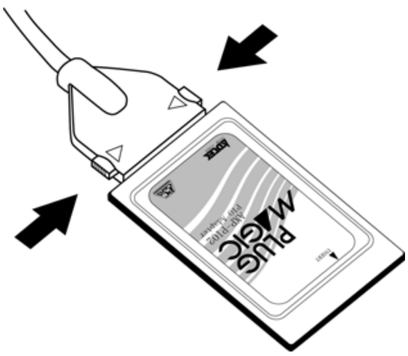
■折り曲げ厳禁。破損してしまったカードは修理できません。



1. カードの取り付け

1-1. ケーブルをカードに取り付ける

カードの上面（PLUG MAGICの文字が見える側）と接続ケーブルのカード側コネクタの上面（図のように△のマークのある側）を合わせ、矢印の方向に静かに差し込みます。ロッキングの金具の「カチッ」という音がすることを確認します。

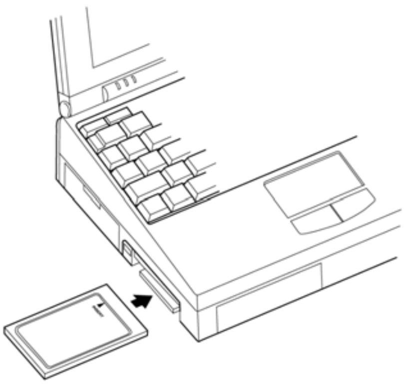


接続ケーブルを無理に曲げたり、コネクタとカードとの接続部に無理な力を加えると動作不良や故障の原因になります。

1-2. パソコンへカードを取り付ける

パソコンのカード・スロットにカードを差し込みます。

カードのインターフェース・コネクタ側をパソコンのPCカードスロットに静かに差し込みます。



PCカードTYPE I スロットには入りません。

PCカードは、誤挿入防止構造になっていますが、無理に差し込もうとすると、パソコンのPCカードスロットやPCカード本体の故障の原因となります。

パソコンの機種によっては、PCカードの裏面を上にし、実装するタイプがあります。ご注意ください。

■カードの取り出し方

PCカードをパソコンから取り出す時は、パソコンのカード・イジェクト・ボタンを押します。カードが少し飛び出します。飛び出した部分を持ち静かに引き抜きます。

ご使用ノートパソコンの取扱説明書カードスロットの項もお読みください。

2. ソフトウェアの組み込み

本製品をご使用になる前に、ソフトウェアの組み込み等の準備が必要です。

ソフトウェアは、サポートソフト（添付サポートディスクまたは弊社ホームページ http://www.adtek.co.jp/ からダウンロード）に収められています。
ここでは、サポートソフトを、フロッピーディスク（以下「サポートディスク」）にコピーして使用する場合について示しています。CD-R 等他のメディアをご使用の場合は、適宜読み替えて作業を進めてください。

以下に本カードを MS-DOS で使用する場合、Windows95/98/Me で使用する場合、Windows2000/XP で使用する場合、それぞれについての手順を説明します。

2-1. MS-DOS で使用する場合

ドライバの組み込み方法

MS-DOSのCONFIG.SYSファイルに、以下の行を追加してください。ただし、カードサービス組み込み行の後に追加しなければなりません。カードサービスの組み込みについては、カードサービスのマニュアルをご覧ください。

例）DEVICE=X:¥PATH¥AXPAD02.EXE[/W:300][I:5]

[ ]のオプションは省略可能です

ドライバのオプションについて

■デバイスドライバのオプションは、以下の通りです。

I/Oアドレスの指定　　「/W:????」
値は16進数で指定してください。

割り込みの指定　　「/I:?」
値は10進数で指定してください。

注）現状のドライバでは１枚のみ組み込み可能です。

上記のI/Oウインドウ及び割り込みの指定は、以下に示すI/Oウインドウ、割り込みレベルを指定するようにしてください。

	ｴﾝﾄﾘ	ウィンドウ(/W1)		
＊	0	0300h	－	0307h
＊	1	0710h	－	0717h
＊	2	02D0h	－	02D7h
＊	3	03D0h	－	03D7h

オプション指定が無い場合、＊のいずれかに設定されます。

割り込み(/I)　　0～15

NEC PC98シリーズ使用の場合は必ずオプションで空いているウィンドウを指定してください。
例）02D0h－02D7

Windows3.1でご使用になる場合は、MS-DOSと同じ手順でドライバを組み込んでください。

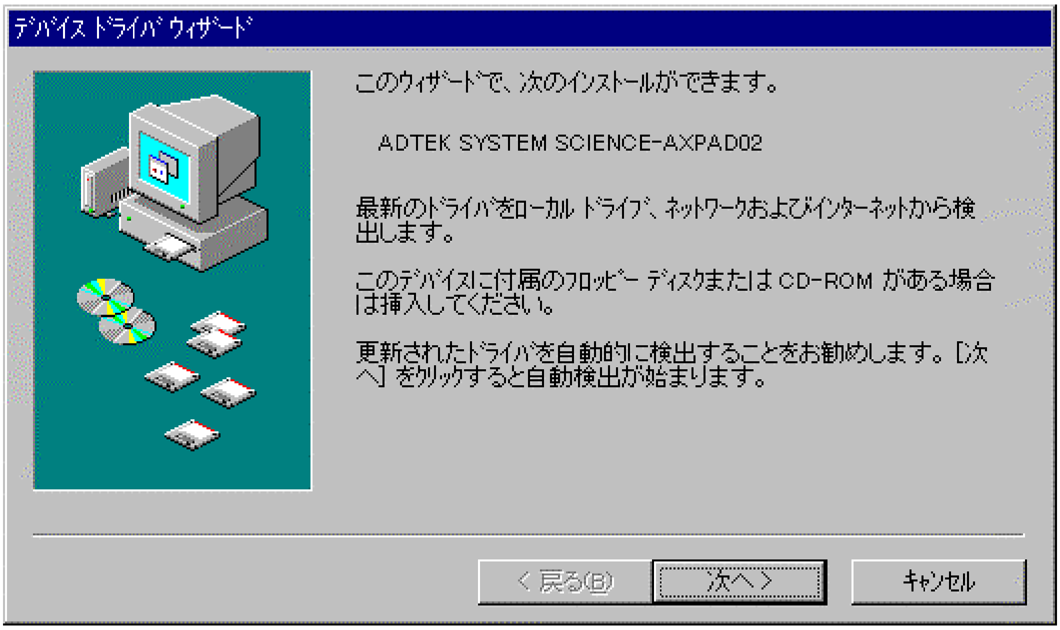
2-2. Windows95/98/Me で使用する場合

2-2-1. カードの登録（インストール）

以下は初めて本カードをご使用いただくとき、もしくは登録の削除をした場合の設定です。
この設定は、1度行うと、次回から登録の削除を行わない限り有効です。

ここではAXP-AD02をWindows95で使用する場合を示します。Windows98/Meで使用する場合は、画面の指示に従って適宜読み替えてください。
Windows95が起動したら、PCカードスロットにAXP-AD02を挿入してください。

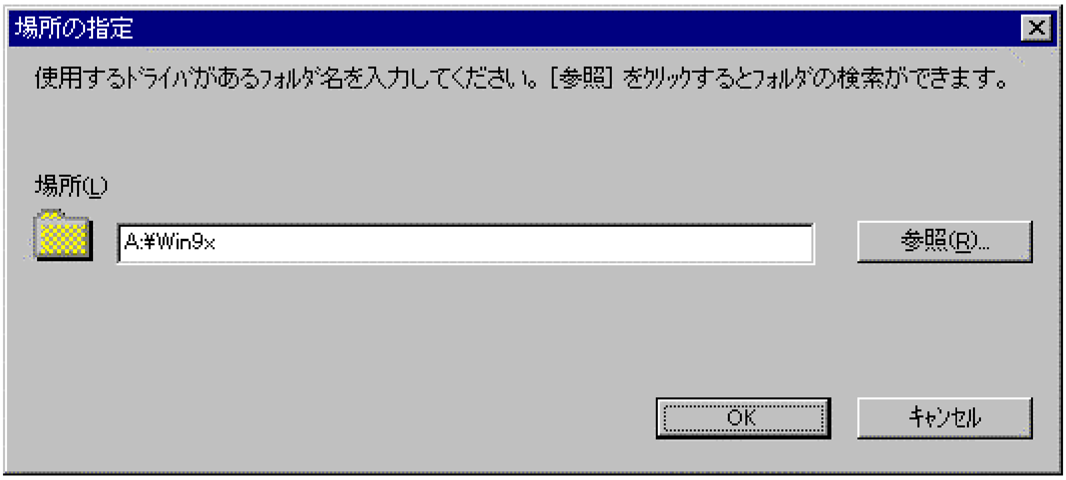
挿入後、以下の画面が表示されますので、「次へ>」をクリックしてください。



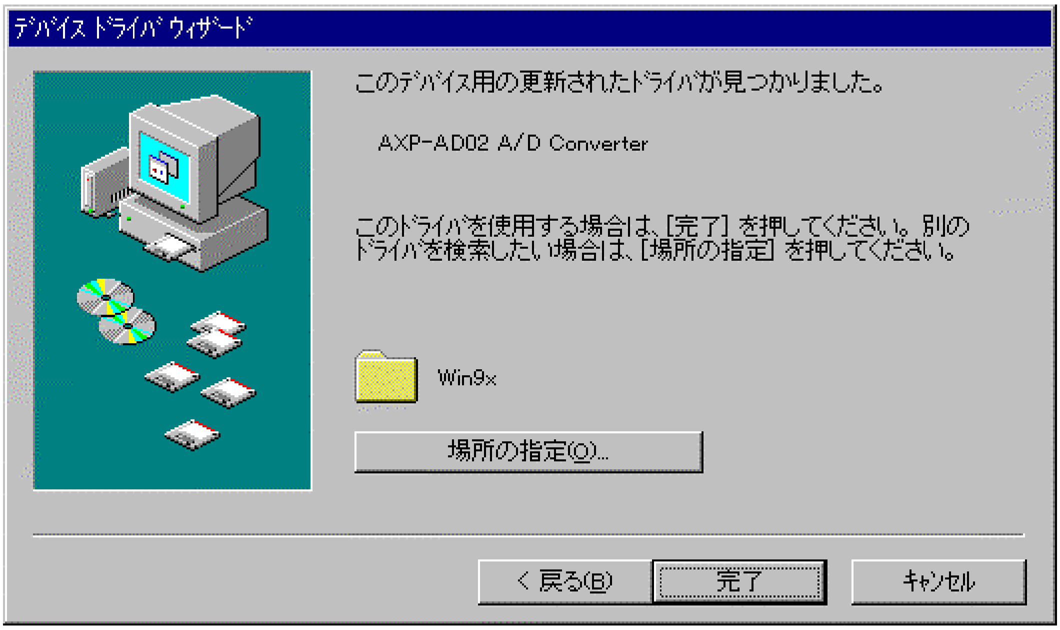
以下の画面が表示されましたら、「場所の指定(O)」をクリックしてください。



サポートディスクをフロッピーディスクドライブに差し込んでください。
ドライブがAドライブの場合は以下のように入力し、「OK」をクリックしてください。



ドライバが見つかったことを確認しましたら、「完了」をクリックしてください。



これでインストール作業は終了です。
数秒後に通常PCカードが挿入されたことを示す認識音（BEEP音）が出ると終了です。
この後、登録の確認を必ず行ってください。

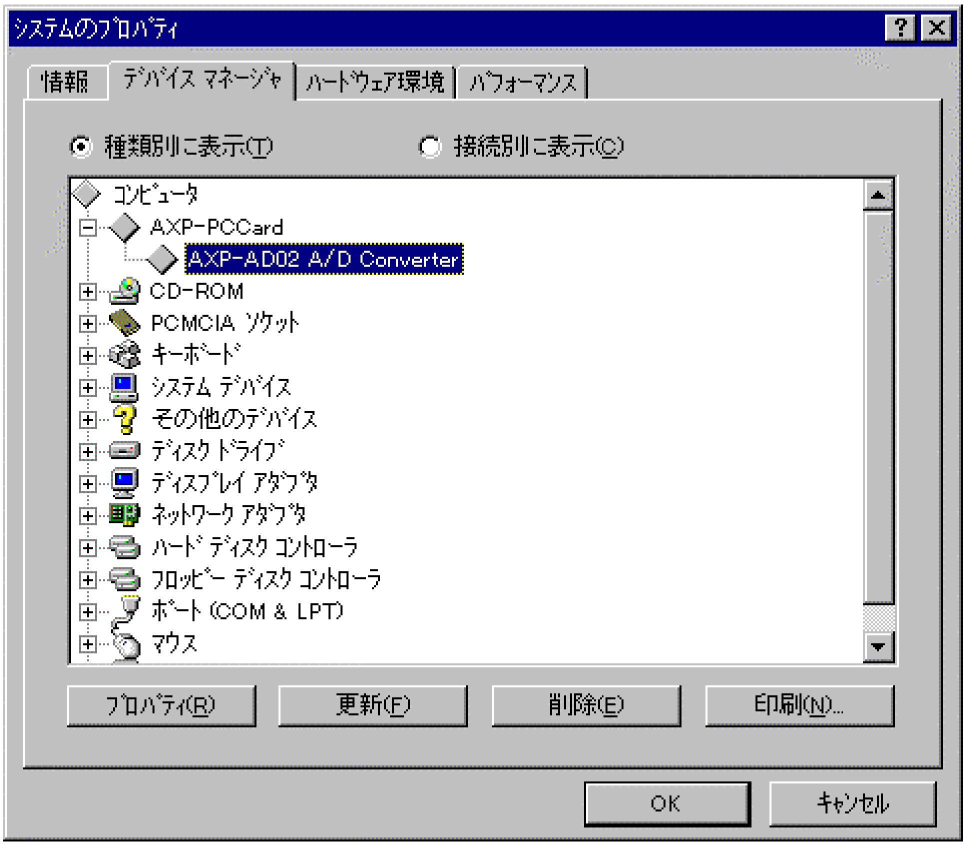
上手くインストールできなかった等、トラブルシューティングはサポートディスクのヘルプおよびreadme.txtをご覧ください。

2-2-2. 登録（インストール）の確認

カードの登録の確認を行います。
コントロールパネルを開き、「システム」をダブルクリックします。



「デバイス マネージャ」のタグをクリックしてください。
「AXP-PCCard」をクリックして、「AXP-AD02 A/D Converter」をダブルクリックしてください。



プロパティが表示されましたら、「リソース」のタグをクリックしてください。



画面が表示されましたら、“競合するデバイス”を確認してください。
図のように“競合なし”であれば、正常に設定が完了しています。



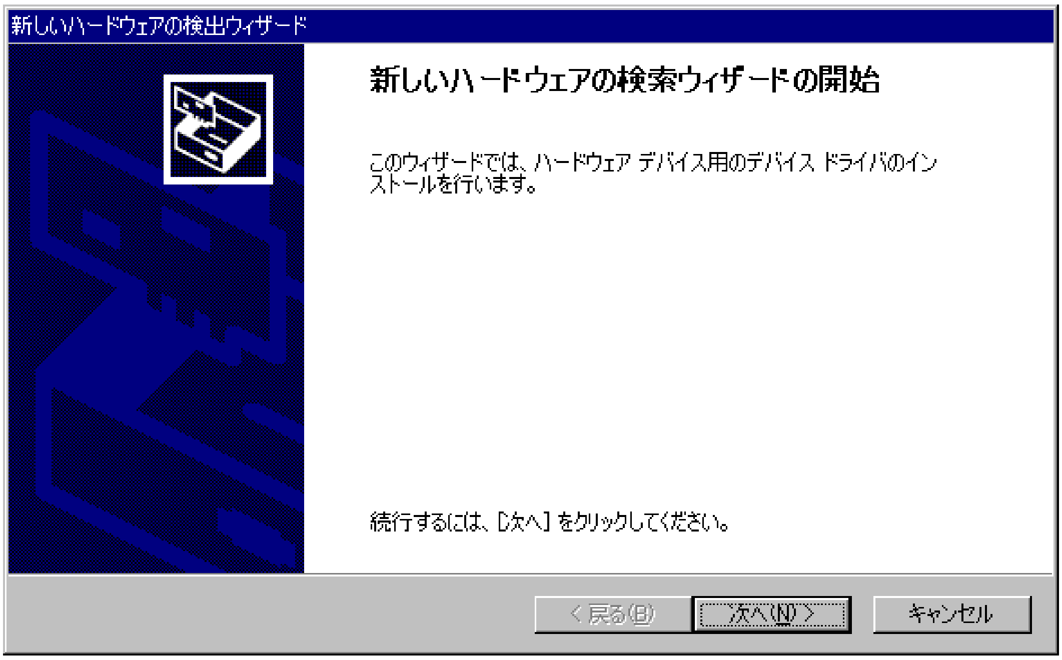
2-3. Windows2000/XP で使用する場合

2-3-1. カードの登録（インストール）

以下は初めて本カードをご使用いただくとき、もしくは登録の削除をした場合の設定です。
この設定は、1度行うと、次回から登録の削除を行わない限り有効です。

ここではAXP-AD02をWindows2000で使用する場合を示します。WindowsXPで使用する場合は、画面の指示に従って適宜読み替えてください。
Windows2000が起動したら「Administrator」でログオンし、PCカードスロットにAXP-AD02を挿入してください。

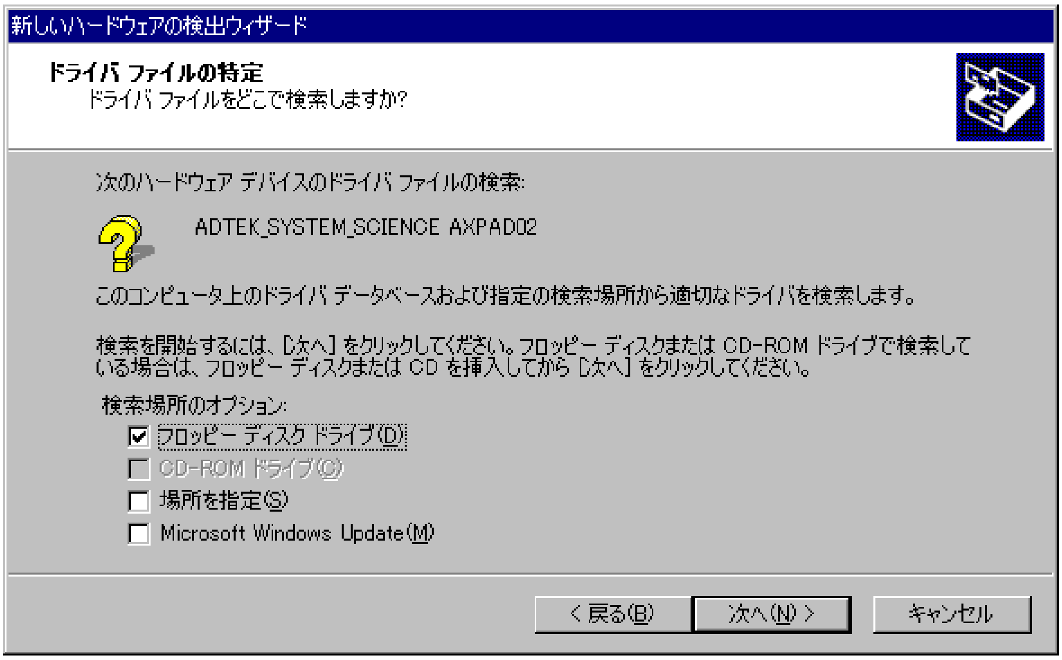
挿入後、以下の画面が表示されますので、「次へ(N)＞」をクリックしてください。



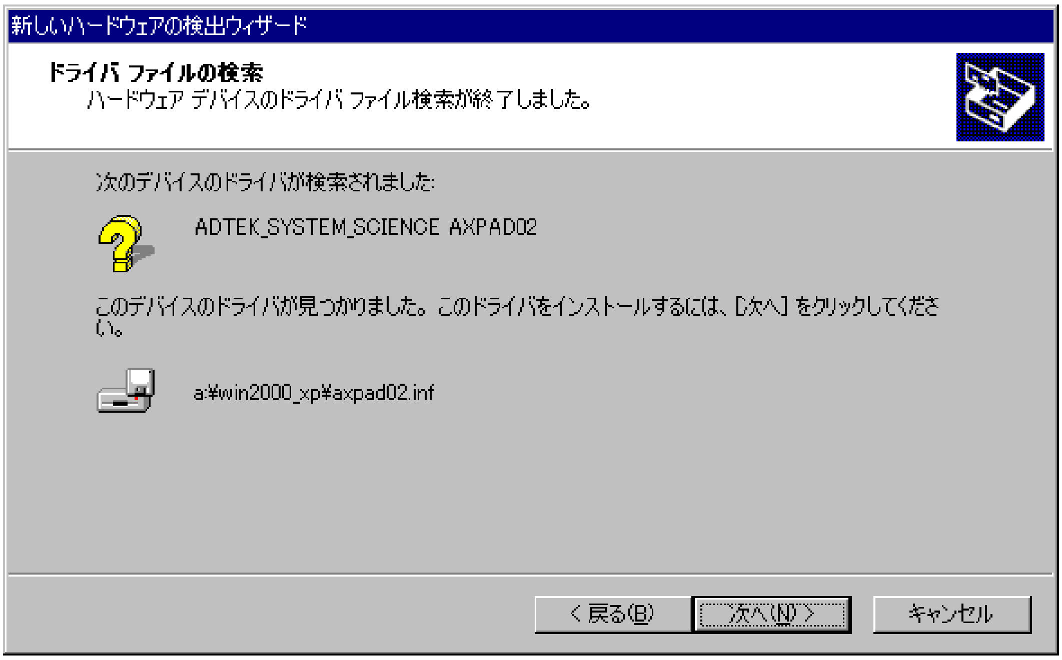
「デバイスに最適なドライバを検索する(推奨)(S)」を選択して、「次へ(N)＞」をクリックしてください。



サポートディスクをフロッピーディスクドライブに差し込んでください。
「フロッピー ディスク ドライブ(D)」を選択して、「次へ(N)＞」をクリックしてください。



「次へ(N)＞」をクリックしてください。



インストールが終了したことを確認しましたら、「完了」をクリックしてください。



これでインストール作業は終了です。
数秒後に通常PCカードが挿入されたことを示す認識音（BEEP音）が出ると終了です。
この後、登録の確認を必ず行ってください。

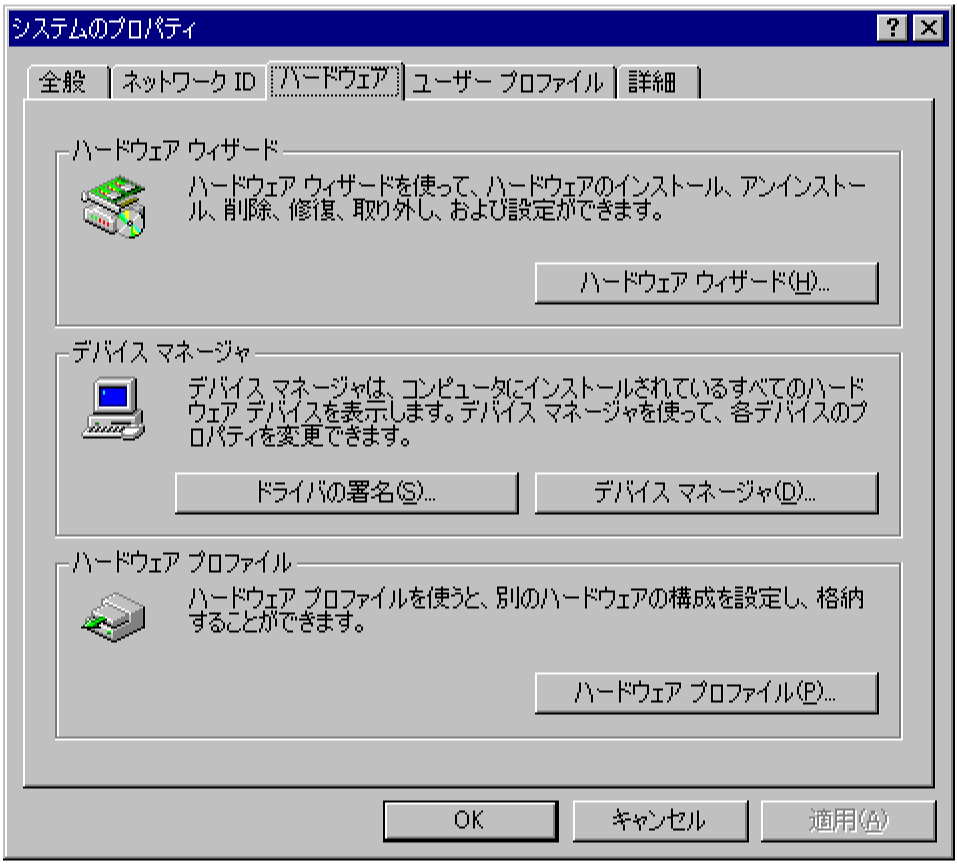
上手くインストールできなかった等、トラブルシューティングはサポートディスクのヘルプおよびreadme.txtをご覧ください。

2-3-2. 登録（インストール）の確認

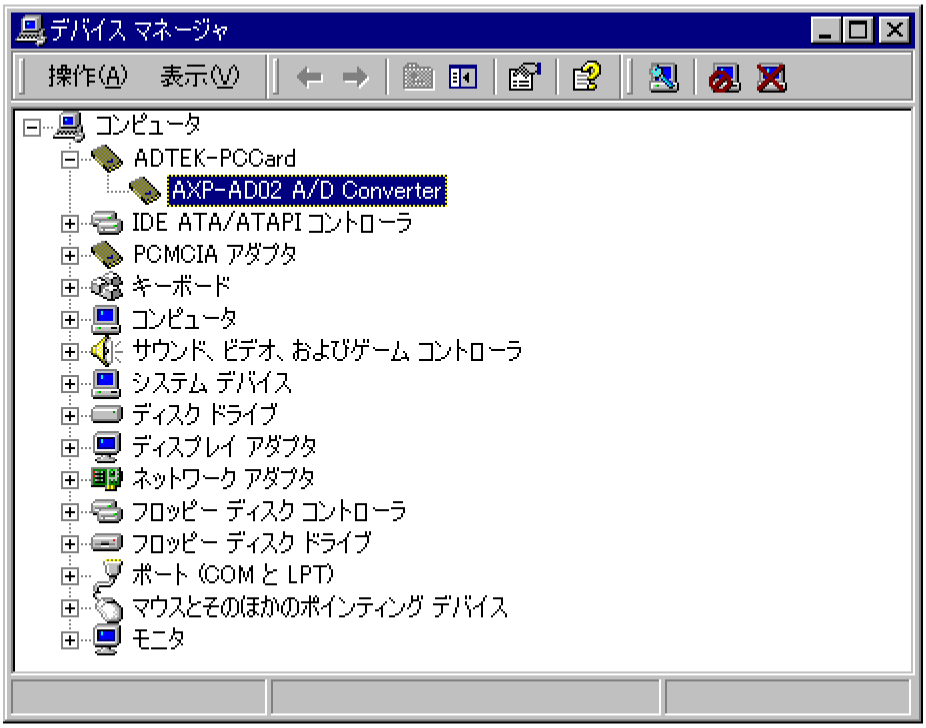
カードの登録の確認を行います。
コントロールパネルを開き、「システム」をダブルクリックします。



「ハードウェア」のタグをクリックして、「デバイス マネージャ(D)」をクリックしてください。



「ADTEK-PCCard」をクリックして、「AXP-AD02 A/D Converter」をダブルクリックしてください。



プロパティが表示されましたら、「リソース」のタグをクリックしてください。



画面が表示されましたら、“競合するデバイス”を確認してください。
図のように“競合なし”であれば、正常に設定が完了しています。



3. アプリケーションの作成

アプリケーションプログラムの作成方法を解説します。
AXP-AD02 は、MS-DOS 用ドライバ、Windows95/98/Me 用ドライバ、Windows2000/XP 用ドライバおよび専用ライブラリ等が付属しています。
これらのファイルは、サポートソフトに収めてあります。ドライバ以外のファイルは、作業環境に合わせてコピーしてご使用ください。
また、サポートソフトには、ドライバのアクセス方法や、実際に動作するサンプルプログラムのソースコードも含まれています。
アプリケーション作成の際に参考にしてください。

3-1. サポートソフトの内容

```
¥
├─ Dos                      MS-DOS用ドライバ/サンプルプログラム
│   ├─ dosv                    DOS/Vマシン用
│   │   ├─ TURBOC                 TURBO/BORANDC用のライブラリ
│   │   ├─ MSC                    MSC用のライブラリ
│   │
│   ├─ pc98                    PC98用
│   │   ├─ TURBOC                 TURBO/BORANDC用のライブラリ
│   │   ├─ MSC                    MSC用のライブラリ
│   │
│   ├─ sample                  参考プログラム
│   │   ├─ ad02lib.h              AXP-AD02ライブラリのヘッダーファイル
│   │   ├─ single.c               シングルサンプリング
│   │   ├─ timer.c                タイマーサンプリング
│   │   ├─ graph.c                タイマーサンプリングでグラフ表示
│   │   ├─ calib.c                校正
│   │
│   ├─ driver
│   │   ├─ axpad02.exe            MS-DOS用ドライバ
│   │   ├─ opendat.exe            カード状態表示ユーティリティ
│   │
│   ├─ readme                  関数ライブラリの説明
│
```

≪次ページへ≫

≪前ページより≫

```
├─ Win9x                    Windows95/98/Me用32bitドライバ
│  ├─ axpad02.inf              インストールファイル
│  ├─ axpad02.vxd              ドライバ
│  ├─ axpad02.dll              DLL
│
├─ Win2000_XP               Windows2000/XP用32bitドライバ
│  ├─ axpad02.inf              インストールファイル
│  ├─ axpad02.sys              ドライバ
│  ├─ axpad02.dll              DLL
│
├─ Sample                   Windows95/98/Me/2000/XP用サンプルソース
│  ├─ Vc                       VCサンプルソース
│  │  ├─ axpad02.h                DLL定義ヘッダ
│  │  ├─ axpad02w.h               ラッパー関数ヘッダ
│  │  ├─ axpad02w.c               ラッパー関数
│  │  ├─ axpad02s.c               ラッパー関数使用例
│  │  ├─ buildvc.txt              サンプルソース構築例
│  │  ├─ Lib
│  │     ├─ Win9x
│  │     │  ├─ axpad02.lib        インポートライブラリ
│  │     │
│  │     ├─ Win2000_XP
│  │        ├─ axpad02.lib        インポートライブラリ
│  │
│  ├─ Vb                       VBサンプルソース
│  │  ├─ axpad02.bas              DLL定義
│  │  ├─ axpad02w.bas             ラッパー関数
│  │  ├─ axpad02s.bas             ラッパー関数使用例
│  │  ├─ buildvb.txt              サンプルソース構築例
│  │
│  ├─ Delphi                   Delphiサンプルソース
│     ├─ axpad02w.pas             DLL定義/ラッパー関数
│     ├─ axpad02s.pas             ラッパー関数使用例
│     ├─ axpad02s.dfm             フォームファイル
│     ├─ builddp.txt              サンプルソース構築例
│
├─ axpad02.hlp/cnt          ヘルプファイル
├─ axpad02.exe              動作チェックアプリケーション
├─ axpad02ap.txt            API仕様書
├─ readme.txt               リードミー
```

3-2. I/O マップ

本カードのI/Oマップ及び、レジスタについて解説します。

3-2-1. I/O マップ

	ライトポート	リードポート	I/O アドレス
1	未使用	サンプリングバッファ	XX0h
2	未使用	ステータス１	XX1h
3	モードセット１	ステータス２	XX2h
4	モードセット２	未使用	XX3h
5	タイムセット	未使用	XX4h
6	スタートコマンド	未使用	XX5h
7	汎用出力	汎用入力	XX6h
8	エンドチャンネル	未使用	XX7h

注）未使用のライトポートにはアクセスしないでください。
　　もし間違えてアクセスした場合は本カードを再挿入してください。

3-2-2. レジスタの説明

○サンプリングバッファ

このバッファはサンプリングデータ読み出し専用で、1024バイトサンプル数で512データ有ります。読み出しは２回１データで１バイト目が上位、２バイト目が下位です。

DB7 … DB0

0	0	0	0	12	11	10	9

常に"0"（DB7〜DB4）

１回目リード

DB7 … DB0

8	7	6	5	4	3	2	1

２回目リード

○ステータス１

このレジスタはサンプリング時のチャンネルの設定状態及び、アクセス状態などを示すレジスタです。リセット時（パワーオン時）00hです。

DB7 … DB0

A2	A1	A0	SA2	SA1	SA0	ADBUSY	SBUSY

SBUSY：カードのシステムビジー

ADBUSY：内部 A/D ビジー

SA2〜SA0：オートインクリメント時の最終チャンネル

A2〜A0：
シングルサンプリング時：サンプリングチャンネルを示す
オートインクリメント時：次にサンプリングするチャンネルを示す

○ステータス２

このレジスタは割り込みステータス及び、サンプリングバッファの状態を示します。
リセット時（パワーオン時）10hです。

DB7							DB0
INTEN	BF-F	BH-F	BE-F	COMEND	BH	EXIN	BF

（割り込みステータス）

BF : バッファオーバーフロー
EXIN : IN2 外部入力
BH : バッファ 256 データ有り
COMEND : コマンド終了

BE-F : バッファエンプティ
BH-F : バッファ 256 データ有り
BF-F : バッファオーバーフロー

割り込み許可

本カードで使用しているバッファは非同期読み出し可能なFIFOメモリです。このバッファはサンプリングモードにより下記に記述する動作の違いがあります。

・シングルサンプリング : オーバーフロー（BF-F）時、古いデータから上書きされます。
・タイマーサンプリング : オーバーフロー時強制終了します。そのため本カードでは、データが256データ分格納された時点で割り込みを発生する事が可能です。また、割り込みを使用しない場合でもBH-F（DB5）によっても確認できます。
特に速いタイムレートでサンプリングを行う場合、本割り込みを使用することをおすすめします。

割り込みフラグステータスについて

割り込みを使用している場合必ず本レジスタを読み込んでください。読み込まない場合、以降の割り込みは発生しなくなります。また、本ステータスは読み込むことでクリアされます。

各ステータスビットについて

・COMEND、BH、EXIN、BF（DB3～DB0）は割り込み用フラグですが、割り込みを使用しない場合でも状態確認用ステータスとして使用可能です。
・BF-F、BH-F、BE-F（DB6～DB4）は1024バイト（A/Dサンプル数512データ）のバッファ状態を示します。また本ビットにおいてはバッファICの状態を直接受けているため、本レジスタ読み出しによるビットクリア動作はありません。そのため本カードリセット直後に本レジスタを読み出すと、10h（BE-F）の値が読み出されます。
・INTEN（DB7）はモードセット２レジスタのINT（DB5）により割り込み許可状態に設定されている場合”1”となります。

〇モードセット１

基本的な動作指示はこのレジスタで行います。リセット時（パワーオン時）はサンプリングチャンネル”0”、シングルサンプリングモードの設定になります。

DB7 … DB0

A2	A1	A0	SRES	BUFFCL	AINC	SS/ST	0

- 0（DB0）：このビットは常に”0”としてください
- SS/ST：“0”：シングルサンプリング
  “1”：タイマーサンプリング
- AINC：“0”：A2～A0 で指定されたチャンネル固定
  “1”：SA2～SA0 と A2～A0 間で指定されたチャンネルをインクリメント
- BUFFCL：“1”：バッファクリアモード
- SRES：“1”：システムリセット
- A2～A0：シングルサンプリング時：固定アドレス
  タイマーサンプリング時：スタートアドレス

SRES（DB4）について

本レジスタおよびバッファメモリ（FIFOメモリ）、A/Dデバイスはリセット対象外となります。バッファメモリを含めカードをリセットする場合は、本ビットとBUFFCL（DB3）のビットを”1”（18h）としてスタートコマンドを実行します。実行後は本レジスタを00hに戻します。

チャンネル切り替え時の注意

本カードの最小サンプリングタイムは10μsですが、チャンネル切り替えを行いながらサンプリングを行う場合、チャンネル切り替え後60μsのディレーを必要とします。そのためタイマーサンプリングとチャンネルオートインクリメントを併用する場合、タイムレートは60μsが最小値となりますので注意してください。

動作モードについて

・シングルサンプリング：サンプルコマンドを書き込むごとに１回サンプリングを実行します。

BUFFCL（DB3）を”1”にした状態で上記モードを実行した場合、サンプルコマンド書き込みタイミングにバッファがクリアされ、サンプル終了後新規データがバッファにセットされます（バッファなしモード）。

・タイマーサンプリング：１回のサンプルコマンド書き込みにより、タイムセットレジスタにセットされた時間間隔でサンプリングを開始します。その場合BUFFCLは必ず”0”としてください。サンプリングの停止はSS/ST（DB1）をシングルサンプリング”0”にすることで停止します。

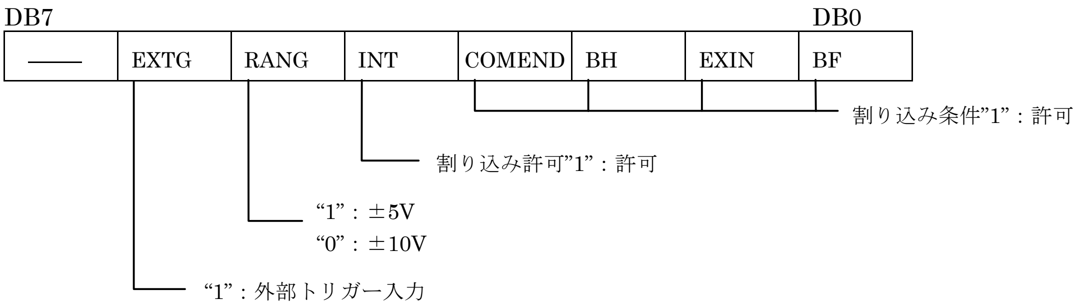
・オートインクリメント：この機能はAINC（DB2）の状態により変わります。またこの機能は上記動作モードに関係なく機能します。

AINC=”0”　A2～A0に指定されたチャンネル固定になります。

AINC=”1”　A2～A0に指定されたチャンネルからエンドチャンネルレジスタSA2～SA0で指定されたチャンネル間を自動でインクリメントループします。

○モードセット２

このレジスタは割り込みの許可および、入力レンジなどの設定を行います。リセット時（パワーオン時）入力レンジは±10V、外部トリガー禁止、割り込み禁止の設定です。



RANG（DB5）について

このビットは出荷時の調整データと連動して設定する必要があり、弊社より供給（サポートソフト内）の関数をご使用ください。また本カードの起動時においても調整データを設定する必要があります。

注）イニシャル時および入力レンジ切り替えの際は必ずサポートソフト内の下記関数を使用してください。

DOS版

イニシャル時　　　　　：関数名　AD02_init( )

入力レンジ切り替え時：関数名　AD02_cal_prset( )

Windows版

イニシャル時　　　　　：関数名　Ad02Create( )

入力レンジ切り替え時：関数名　Ad02SetMode( )

EXTG（DB6）について

外部トリガー設定時汎用入出力は下記の機能になります。

IN1　：トリガー入力　最小パルス幅　　0.5μs

最小パルス周期　10μs

OUT1 ：SBUSY出力になり、”1”でサンプリング中、出力パルスは約8.6μsです。

注）ただしチャンネルオートインクリメントなどのチャンネル切り替えが入る場合、最小パルス周期は60μsとなります。

タイマーサンプリングモード時は最初のパルスのみで、以降の入力は無視されます。

外部トリガー（EXTG）機能

この機能はスタートコマンドを使用せず外部信号入力（汎用入力：IN1）を使用しサンプリングを開始させる機能です。この機能についてもシングル／タイマのいずれのモードでも使用可能です。

シングルモードにおいて外部トリガを停止する場合、本ビットを”0”に設定します。またタイマーモードにおいては、本ビットを”0”にしてからモードセット１レジスタのSS/ST（DB0）をシングルサンプリング”0”にすることで停止します。操作の順番を間違えると余分なデータが取り込まれる場合があります。

○タイムセット

タイマーサンプリング時の時間設定用レジスタです。

DB7　　　　　　　　　　　　　　　DB0

8	7	6	5	4	3	2	1

1～255（10μs～2.55ms）

モードセット２レジスタのSS/STをセットする前に本レジスタをセットするようにしてください。

○スタートコマンド

このレジスタは、セットするデータは有りません。任意のデータを書き込んでください。またモードセットＸレジスタでセットされたモードの実行はすべてこのコマンドを発行することで実行されます。

○エンドチャンネル

チャンネルオートインクリメントの最終チャンネルの設定用レジスタです。
DB5～DB0は必ず”0”にしてください。

DB7　　　　　　　DB5　　　　　　　　　　DB0

SA2	SA1	SA0	0	0	0	0	0

オートインクリメント設定手順
参考設定モード：シングルサンプルオートインクリメントモード
　　　　　　　　スキャンするチャンネルは３～５チャンネル、I/Oウインドウ300h
　　　　　　　　スタートチャンネル３、エンドチャンネル５

```
#define     BBASE       0x300

#define     MOD1REG     BBASE+0x02      //モードセット１
#define     ENDREG      BBASE+0x07      //エンドチャンネル
#define     START       BBASE+0x05      //スタートコマンド

void incmode(void)
{
outportb(MOD1REG,0x18)      //カードシステムリセット
outportb(START,0x0)         //コマンド実行
outportb(MOD1REG,0x60)      //スタートチャンネルセット
                            //この時点ではAINCはセットしません
outportb(ENDREG,0xa0)       //エンドチャンネル
outportb(MOD1REG,0x64)      //ここでAINCをセット
                            //設定終了
outportb(START,0x0)         //サンプルスタート
  ⋮
その他の処理
}
```

○汎用入出力

リセット時（パワーオン時）OUT4～OUT1出力は"0"です。

DB7							DB0
――	――	――	――	IN4/OUT4	IN3/OUT3	IN2/OUT2	IN1/OUT1

汎用入出力のIN2（DB1）は外部割り込み入力として使用可能です。
設定はモードセット２レジスタのINT（DB4="1"）により割り込みを許可状態にし、EXIN（DB1）フラグを"1"にセットします。また、割り込みの有無はステータス２レジスタのEXIN（DB1）に反映されます。割り込み信号入力は立ち上がりエッジで検出されます。
この機能はEXTG（外部トリガ）機能に影響されませんので、本機能との組み合わせで外部からのリモート処理などに有効です。

その他注意事項

・汎用入出力は、カード電源の投入およびパソコン本体リセット直後数十ms間出力が安定しないため、その影響により不都合が生じるような装置においては注意が必要です。
・本カード内部に使用してありますA/Dデバイスは、多用途の条件に対応するため非常にたくさんの機能を持っております。また、I/Oカードという性質の製品であるため、アナログ部の調整などはソフトウェアで行うことになります。
当然本カードにおいても出荷時に調整を行い、そのデータはカード内に持たせております。しかしそのデータは電源の投入だけでは設定されません。また、入力レンジについても同様です。本カードで正確な計測を行うためには、サポートソフト内のライブラリ（関数）により調整データを設定する必要があります。
・本マニュアルにおいて、調整に関する機能についてはハード上での操作方法は記載してありません。その理由としては処理があまりにも煩雑すぎるのと、操作を間違えることによりデータを壊す可能性があるためです。そのためご面倒でもサポートソフト内のライブラリを熟知されると共に、カードのイニシャルおよび入力レンジ切り替えの際はサポートソフト内のライブラリ（関数類）をご使用いただくようお願いします。

3-3. MS-DOS アプリケーションの作成

MS-DOS上で動作するアプリケーションの作成方法を解説します。
実際のプログラミングについては、サポートソフト内のMS-DOS用C言語サンプルソースをご参照ください。

3-3-1. サンプルソース及び関数ライブラリ概要

サンプルソースはサポートソフトの¥DOS¥SAMPLEディレクトリ内に入っています。またサンプルの実行ファイルについてもTURBO C／MSC用が¥DOSVと¥PC98ディレクトリ内にあります。動作確認などにご使用ください。

SINGLE.C	シングルサンプル動作
TIMER.C	タイマーサンプル動作
GRAPH.C	波形表示サンプル（TURABO C／Borland C専用）
CALIB.C	カード校正サンプル

関数ライブラリについての詳細はサポートソフトのDOSディレクトリ内「readme」をご覧ください。

3-3-2. 関数ライブラリ解説補足

本関数ライブラリは、AXP-AD02を最大限活用するために必要な内容を備えておりますが、もし関数の機能に不十分さを感じられた場合を考慮し、本カードはI/Oを一部公開しております。

関数名　AD02_info( )

カードのI/Oアドレスおよび割り込みリソースを取得するための関数です。本カードの制御プログラム作成時に本関数を使用し、リソース情報を取得してください。

関数名　AD02_init( )

この関数は、組み込み済みのドライバからI/Oウィンドウ先頭アドレスや割り込みなどのリソースの取得およびカードの環境データ生成などを行っています。カードイニシャライズ時はこの関数をご使用ください。

関数名　AD02_cal_prset( )

この関数についても出荷時の調整データの操作を行っております。レンジの切り替えは必ずこの関数で行う必要があります。

I/Oモードセット２によるビット操作のみでは不十分です

関数名　AD02_close( )

上記関数を使用する都合上アプリケーションの終了には必ずこの関数も実行してください。

カードの校正関係は以下の関数をご使用ください。

関数名　AD02_cal_control( )
　この関数は基準電源を使用してシステム（PCも含む）全体での校正モードです。

基本的にはこのモードを使用する必要はないはずですが、使用するPC及び環境により必要な場合があります。この校正モードを使用する場合、カードをPC上で30分程度エージングしてから校正を行ってください。

校正の手順はカードのレンジに対するオフセット／ゲイン調整電圧を入力し2、3回交互に繰り返し行ってください。

この校正モードで調整しきれない場合、下記のオフセット校正を行ってください。

関数名　AD02_cal_OFSS_ADJ( )
　この校正モードは基準電源を使用しないでオフセットの微調整をします。調整手順は、
　アナログ入力をGNDに接続し、本関数を実行します。

関数名　AD02_cal_adc_load( )
　AD02_cal_control( )で設定した校正値をカード内部のADCにセットします。またその時
　の校正データは、アプリケーション終了まで有効です。校正データの参照および保存は
　サポートソフトのDOSディレクトリ内「readme」をご覧ください。

校正値を出荷時に戻したい場合はAD02_cal_prset( )で戻せます。

3-4. Windows アプリケーションの作成

Windows上で動作するアプリケーションの作成方法を解説します。
実際のプログラミングについては、サポートソフト内のWindows用サンプルソース（Visual C++、Visual Basic、Delphiに対応）をご参照ください。

3-4-1. サンプリング動作モード設定について

当カードでは複数のサンプリングモードをサポートしております。ここではこれらのサンプリングモードについてWindows用APIを使った例で解説いたします。なお、以下の例はVisual Basicを使用した例ですので予めご了承ください。

1. シングルサンプリング

このサンプリング方式は、AD変換されたデータを１つだけ取得したいときに使用します。サンプリング終了を判定するには、サンプリングステータス情報の取得かEOCメッセージの受信で行います。サンプリングモード設定時に、dwTimer値を”0”に設定すると自動的にこのサンプリング方式になることにご注目ください。

例）

サンプリング対象チャンネル	Ch0
サンプリングデータ数	1
タイマ値	0（シングルサンプリングのため）
トリガ	内部
入力電圧レンジ	±10V

```
Dim md As AD02_MODE
  ⋮
  ⋮
md.dwChFirst = 0                          ' サンプリング開始チャンネル
md.dwChLast = 0                           ' サンプリング最終チャンネル
md.cwBuffer = 1                           ' サンプリングバッファサイズ
md.dwSampleMode = AD02_SM_ONESHOT         ' サンプリングモード
md.dwTimer = 0                            ' タイマ値（シングルなので”0”を指定）
md.dwTrigSource = AD02_TS_INTERNAL        ' トリガソース
md.dwVoltageRange = AD02_VR_S10V          ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)
```

2. タイマーサンプリング（ワンショット）

このサンプリング方式は、指定したデータ数のみタイマーサンプリングを行いたい場合に使用します。サンプリング終了の判定は、サンプリングステータス情報の取得かEOCメッセージの受信で行います。ただし、ドライバの仕様上データ数を256以下に設定した場合、サンプリング終了を判定することができない場合がございますので、必要な数よりも多めにデータ数を指定する必要があることを予めご了承ください。

例）

サンプリング対象チャンネル	Ch0
サンプリングデータ数	512
タイマ値	1（10×1 = 10μs）
トリガ	内部
入力電圧レンジ	±10V

```
Dim md As AD02_MODE
  ⋮
  ⋮
md.cwBuffer = 512                          ' サンプリングバッファサイズ
md.dwChFirst = 0                           ' サンプリング開始チャンネル
md.dwChLast = 0                            ' サンプリング最終チャンネル
md.dwSampleMode = AD02_SM_ONESHOT          ' サンプリングモード
md.dwTimer = 1                             ' タイマ値（10μs）
md.dwTrigSource = AD02_TS_INTERNAL         ' トリガソース
md.dwVoltageRange = AD02_VR_S10V           ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)
```

3. タイマーサンプリング（連続）

このサンプリング方式は、長時間連続してサンプリングを行いたい場合に使用します。ユーザはサンプリングバッファが溢れないように、定期的にデータを取得する必要がございますのでご注意ください。データ取得にはユーザタイマを使用するかバッファハーフフルメッセージの受信といったタイミングで行います。どうしてもサンプリングバッファが溢れる場合は、サンプリングバッファを多く指定することやデータ取得タイミングを速くする等の工夫が必要です。

例）

サンプリング対象チャンネル	Ch0
サンプリングデータ数	∞
タイマ値	10（10×10 = 100μs）
トリガ	内部
入力電圧レンジ	±10V

```
Dim md As AD02_MODE
   ⋮
   ⋮
md.cwBuffer = 65535                             ' サンプリングバッファサイズ
md.dwChFirst = 0                                ' サンプリング開始チャンネル
md.dwChLast = 0                                 ' サンプリング最終チャンネル
md.dwSampleMode = AD02_SM_CONTINUOUS            ' サンプリングモード
md.dwTimer = 10                                 ' タイマ値（100μs）
md.dwTrigSource = AD02_TS_INTERNAL              ' トリガソース
md.dwVoltageRange = AD02_VR_S10V                ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)
```

3-4-2. サンプリング開始トリガについて

当カードは、サンプリング開始の方法として内部トリガで開始する方法と外部トリガで開始する方法がございます。外部トリガの接続方法は「4. 外部信号との接続　4-1. コネクタ信号表」をご参照ください。

1. 内部トリガ指定

内部トリガを指定した場合、サンプリング開始はAd02wStartSampling APIを実行した時点での開始となります。

例）

サンプリング対象チャンネル	Ch0
サンプリングデータ数	∞
タイマ値	10（10×10 = 100μs）
トリガ	内部
入力電圧レンジ	±10V

```
Dim md As AD02_MODE
  ：
  ：
md.cwBuffer = 65535                        ' サンプリングバッファサイズ
md.dwChFirst = 0                           ' サンプリング開始チャンネル
md.dwChLast = 0                            ' サンプリング最終チャンネル
md.dwSampleMode = AD02_SM_CONTINUOUS       ' サンプリングモード
md.dwTimer = 10                            ' タイマ値（100μs）
md.dwTrigSource = AD02_TS_INTERNAL         ' トリガソース
md.dwVoltageRange = AD02_VR_S10V           ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)               ' モード設定
Ad02wStartSampling(wLogSocket)             ' サンプリング開始
```

2. 外部トリガ指定

外部トリガを指定した場合、サンプリング開始はAd02wStartSampling APIを実行した後、外部からのトリガが発生した時点での開始となります。

例）

サンプリング対象チャンネル	Ch0
サンプリングデータ数	∞
タイマ値	10（10×10 = 100μs）
トリガ	外部
入力電圧レンジ	±10V

```
Dim md As AD02_MODE
  ⋮
  ⋮
md.cwBuffer = 65535                          ' サンプリングバッファサイズ
md.dwChFirst = 0                             ' サンプリング開始チャンネル
md.dwChLast = 0                              ' サンプリング最終チャンネル
md.dwSampleMode = AD02_SM_CONTINUOUS         ' サンプリングモード
md.dwTimer = 10                              ' タイマ値（100μs）
md.dwTrigSource = AD02_TS_EXTERNAL           ' トリガソース
md.dwVoltageRange = AD02_VR_S10V             ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)                 ' モード設定
Ad02wStartSampling(wLogSocket)               ' 外部トリガ受付状態
```

3-4-3. 入力電圧レンジについて

モード設定時に入力電圧レンジを設定することにより±10Vか±5Vのどちらかを選択することが可能です。

1. ±10V 指定

例）シンプルサンプリング

サンプリング対象チャンネル	Ch0
サンプリングデータ数	1
タイマ値	0（シングルサンプリングのため）
トリガ	内部
入力電圧レンジ	±10V

```
Dim md As AD02_MODE
  ⋮
  ⋮
md.cwBuffer = 1                          ' サンプリングバッファサイズ
md.dwChFirst = 0                         ' サンプリング開始チャンネル
md.dwChLast = 0                          ' サンプリング最終チャンネル
md.dwSampleMode = AD02_SM_ONESHOT        ' サンプリングモード
md.dwTimer = 0                           ' タイマ値（シングルなので"0"を指定）
md.dwTrigSource = AD02_TS_INTERNAL       ' トリガソース
md.dwVoltageRange = AD02_VR_S10V         ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)
```

2. ±5V 指定

例）シンプルサンプリング

サンプリング対象チャンネル	Ch0
サンプリングデータ数	1
タイマ値	0（シングルサンプリングのため）
トリガ	内部
入力電圧レンジ	±5V

```
Dim md As AD02_MODE
  ⋮
  ⋮
md.cwBuffer = 1                          ' サンプリングバッファサイズ
md.dwChFirst = 0                         ' サンプリング開始チャンネル
md.dwChLast = 0                          ' サンプリング最終チャンネル
md.dwSampleMode = AD02_SM_ONESHOT        ' サンプリングモード
md.dwTimer = 0                           ' タイマ値（シングルなので"0"を指定）
md.dwTrigSource = AD02_TS_INTERNAL       ' トリガソース
md.dwVoltageRange = AD02_VR_S5V          ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)
```

3-4-4. チャンネルインクリメントサンプリングについて

当カードはADCが８チャンネルありますので、チャンネルインクリメントサンプリングが可能です。例えば、サンプリング開始チャンネルに０、最終チャンネルに３を設定した場合、サンプリングは 0･1･2･3･0･1･2･3･0･1･2･3･…のような順番で行われます。サンプリングバッファサイズには実際に必要なデータ数×チャンネル数を指定していることにご注目ください。なお、チャンネルの切り替えには60μ秒以上のディレーを必要としますのでタイマ値の設定にご注意ください。

例）

サンプリング対象チャンネル	Ch0、Ch1、Ch2、Ch3
サンプリングデータ数	512×4 = 2048（512／Ch）
タイマ値	10（10×10 = 100μs）
トリガ	内部
入力電圧レンジ	±10V

```
Dim md As AD02_MODE
  ⋮
  ⋮
md.cwBuffer = 512*4                     ' サンプリングバッファサイズ
md.dwChFirst = 0                        ' サンプリング開始チャンネル
md.dwChLast = 3                         ' サンプリング最終チャンネル
md.dwSampleMode = AD02_SM_ONESHOT       ' サンプリングモード
md.dwTimer = 10                         ' タイマ値（100μs）60μs以上を指定
md.dwTrigSource = AD02_TS_INTERNAL      ' トリガソース
md.dwVoltageRange = AD02_VR_S10V        ' 入力電圧レンジ

Ad02wSetMode(wLogSocket, md)
```

3-4-5. サンプリングステータス監視について

当カードのプログラミングでサンプリング終了時の判定は、ドライバからのメッセージを受けて判断する方法が一般的ですが、ステータス監視ループを行ってサンプリング終了待ちをする方法もございます。この場合、ループ処理の中に他のプロセスにも制御を渡す処理が必要になります。また、処理方法によってはCPUの占有率が100%近くになり、ドライバを含む他の制御に影響を与えることがありますので十分にご注意ください。
なるべくユーザタイマを使ってのステータス監視やドライバからのメッセージをトリガにするような処理になるようにご設計ください。

例）CPUを100%近く占有してしまう例

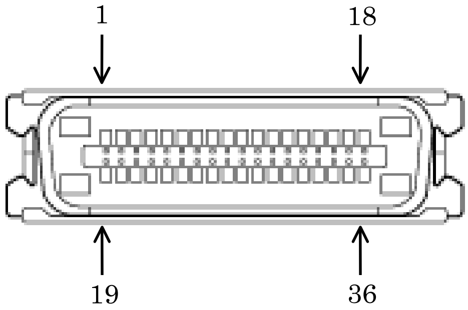
```
  Ad02wStartSampling ( wLogSocket )
  Do While( True )
      Ad02wGetSamplingStatus( wLogSocket, dwNumData, dwStatus )
      If dwStaus And AD02_BMSST_EOC Then ExitDo
  Loop
```

4. 外部信号との接続

カードと外部機器との接続は、添付の付属ケーブルを使用して接続ケーブルの入出力コネクタに接続するか、もしくは市販のコネクタを利用して接続します。
簡単な実験などを行う場合は付属ケーブルの入出力端子に IC クリップなどを取り付けると良いでしょう。

4-1. コネクタ信号表

ﾋﾟﾝ番号	信号名	ﾋﾟﾝ番号	信号名
1	CH1	19	AGND
2	CH2	20	AGND
3	CH3	21	AGND
4	CH4	22	AGND
5	CH5	23	AGND
6	CH6	24	AGND
7	CH7	25	AGND
8	CH8	26	AGND
9	DGND	27	IN1
10	IN2	28	IN3
11	IN4	29	OUT1
12	OUT2	30	OUT3
13	OUT4	31	NC
14	NC	32	NC
15	NC	33	NC
16	NC	34	NC
17	NC	35	NC
18	NC	36	NC



■接続ケーブル側コネクタ
DDK 製 57F-36 又は相当品
■上記に適合する受け側コネクタは
以下の型名のものが使用できます。
○フラットケーブル圧着型
DDK 製 57F-40360-20S
○ソルダカップ型
DDK 製 57-60360

■兼用ピンの説明
OUT1 ：SBUSY　　システムビジーフラグ
IN1　 ：EXTG　　 外部トリガ入力

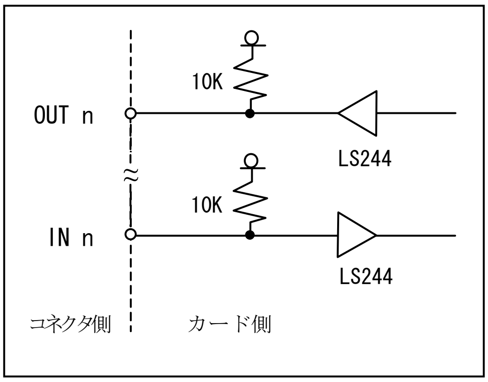
注）外部トリガモード設定時に有効です。

■その他
IN2：外部入力割り込み

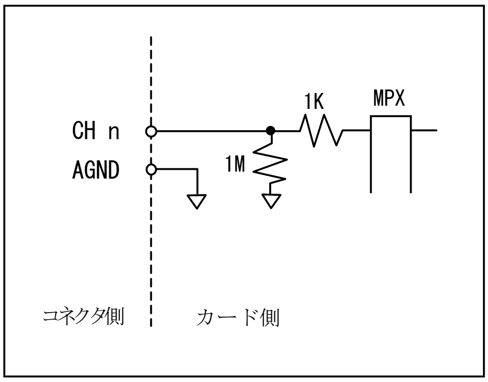
AGNDとDGNDはカード内部で接続されています。

4-2. 入出力信号等価回路

汎用入出力には特別な保護回路などはついておりません。よって過大電圧を加えたり、出力信号をGNDや他の信号と短絡（ショート）するとカードの故障の原因となります。

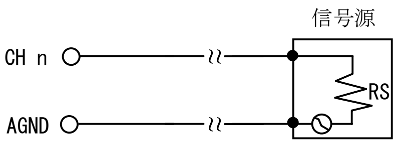


汎用入出力

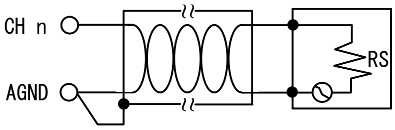


アナログ入力

4-3. 信号接続例



フラットケーブル時



シールドケーブル時

AGNDは、各チャンネルに対応して1本ずつあります。

すべてのAGNDは、カード内部基板上で共通になっております。そのため、異種レベルのAGNDの信号源を同時に使用することはできません。

5. 製品仕様

①A/D変換方式	：逐次比較変換方式
②A/D変換分解能	：12ビット
③A/D変換速度	：10μs
	注）チャンネル切り替え時は60μsのディレーが必要です。
④入力抵抗	：1MΩ
	（アナログーグランド間　1MΩでプルダウン処理）
⑤入力方式	：シングルエンド入力
⑥入力チャンネル数	：８チャンネル
⑦入力レンジ	：±10V、±5V（ソフト設定）
⑧入力最大電圧	：±33V（アナログーグランド間）
⑨変換精度	：積分非直線性誤差　±1LSB（TYP）
	±2LSB（MAX）
	：微分非直線性誤差　±1LSB（TYP）
	±2LSB（MAX）
	注）両レンジにおいて
	：オフセット誤差　±1LSB（TYP）
	：フルスケール誤差　±1LSB（TYP）
	注）両レンジにおいて、キャリブレーションを行った上でのデータです。
	：温度特性　オフセット　35ppm／℃
	フルスケール　139ppm／℃
⑩変換コード	：オフセットバイナリ
⑪サンプリング機能	：一定周期サンプリング（10μs～2.55ms）
	：チャンネルオートインクリメント
⑫非同期FIFO	：512ワード
	注）データ読み出しはバイト単位です。
⑬割り込み機能	：バッファフル割り込み／ハーフバッファ割り込み／システムBUSY割り込み／外部入力割り込み
⑭汎用入出力	：各４ビットTTLレベル入出力
	（入力１点のみ割り込み可能　IN2）
⑮外部トリガー	：トリガー入力／BUSYフラグ出力
	注）汎用入出力と兼用です。
⑯電源電圧	：＋5V±5％単一
⑰消費電流	：234mA（MAX）、133mA（TYP）
⑱動作温度範囲	：＋5℃～＋35℃
⑲保存温度	：－15℃～＋70℃

製品のメンテナンスについて

- ◆ ハードウェア製品の故障修理やメンテナンスなどについて、弊社—株式会社アドテックシステムサイエンスでは、製品をお送りいただいて修理／メンテナンスを行い、ご返送する、センドバック方式で承っております。

- ◆ 保証書に記載の条件のもとで、保証期間中の製品自体に不具合が認められた場合は、その製品を無償で修理いたします。保証期間終了後の製品について修理が可能な場合、又は改造など保証の条件から外れたご使用による故障の場合は、有償修理となりますのであらかじめご了承ください。

- ◆ 修理やメンテナンスのご依頼にあたっては、保証書を製品に添え、ご購入時と同程度以上の梱包状態に『精密部品取扱注意』と表示のうえお送りください。また、ご送付されるときは、製品が迷子にならないよう、前もって受付担当者をご確認ください。製品が弊社に到着するまでの事故につきましては、弊社は責任を負いかねますので、どうか安全な輸送方法をお選びください。

- ◆ 以上の要項は日本国内で使用される製品に適用いたします。日本の国外で製品を使用される場合の保守サービスや技術サービス等につきましては、弊社の各営業所にご相談ください。

製品のお問い合わせについて

◆ お買い求めいただいた製品に対する次のようなお問い合わせは、お求めの販売店または株式会社アドテックシステムサイエンスの各営業所にご連絡ください。

・お求めの製品にご不審な点や万一欠品があったとき
・製品の修理
・製品の補充品や関連商品について
・本製品を使用した特注製品についてのご相談

◆ 技術サポート ——— 技術的な内容のお問い合わせは、「ファックス」「郵送」「E-mail」のいずれかにて、下記までお問い合わせください。また、お問い合わせの際は、内容をできるだけ詳しく具体的にお書きくださいますようお願いいたします。

———— 技術的な内容のお問い合わせ先 ————

株式会社 アドテック システム サイエンス　テクニカルサポート
〒240-0005
神奈川県横浜市保土ヶ谷区神戸町 134　YBP ウエストタワー 8F
E-mail　support@adtek.co.jp
Fax　045-331-7770

改訂履歴

発行年月日　　1997 年 12 月 10 日　初版

発行年月日　　2002 年 10 月 08 日　第 2 版
Windows アプリケーションの作成方法を追加
Windows へのソフトウェアの組み込み手順を追加

発行年月日　　2003 年 10 月 16 日　第 3 版
ソフトウェアの組み込み手順を修正
製品のメンテナンス・お問い合わせに関する情報を追加

発行年月日　　2005 年 03 月 22 日　第 4 版
本社住所を変更

PLUG MAGICシリーズ A/Dコンバータ

AXP-AD02

取扱説明書

第4版発行　2005年03月22日
発行所　株式会社 アドテック システム サイエンス
〒240-0005　神奈川県横浜市保土ヶ谷区神戸町134
YBPウエストタワー 8F
Tel　045-331-7575 ㈹　　Fax　045-331-7770

AXP-033-050322